SSP – Transport binding

V1.0

WV Internal Tracking Number: WV-015

Notice

Implementation of all or part of any Specification may require licenses under third party intellectual property rights, including without limitation, patent rights (such a third party may or may not be a Supporter). The Sponsors of the Specification are not responsible and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights.

THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN ARE PROVIDED ON AN "AS IS" BASIS WITHOUT WARRANTY OF ANY KIND AND ERICSSON, MOTOROLA and NOKIA DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL ERICSSON, MOTOROLA or NOKIA BE LIABLE TO ANY PARTY FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS, LOSS OF USE OF DATA, INTERRUPTION OF BUSINESS, OR FOR DIRECT, INDIRECT, SPECIAL OR EXEMPLARY, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES OF ANY KIND IN CONNECTION WITH THIS DOCUMENT OR THE INFORMATION CONTAINED HEREIN, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE. The above notice and this paragraph must be included on all copies of this document that are made.

Table of contents

1. References

[RFC0793] "Transmission Control Protocol", Jon Postel, September 1981.
[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", Bradner, S., March 1997.
[RFC2616] “Hypertext Transfer Protocol – HTTP/1.1”, Fielding R.; Gettys J.; Mogul J.; Frystyk H.; Masinter L.; Leach P.; Berners-Lee T., June 1999.

2. Introduction

The SSP messages are carried and transmitted by the reliable HTTP / HTTPS overTCP transport protocol. The physical connections carry the service requests of the Requestor Server and the notification requests of the Provider Server.

The SSP transactions are independent of the underlying transport protocol transactions, i.e. one SSP transaction may be carried by two transport protocol transactions.

The SSP transaction identifier is always generated by the initiator of the transaction request. The SSP response MUST include the same transaction identifier, which was transmitted in the request. The SSP transaction request and response carry the identifier of the service provisioning session.

3. The HTTP / HTTPS over TCP binding

3.1. Connection Pair

The HTTP / HTTPS protocol is an asymmetrical protocol, therefore two physical TCP connections are needed for the HTTP / HTTPS binding. One TCP connection is originated as the HTTP / HTTPS client from the Requestor Server to the Provider Server, i.e the physical connection 1, and similarly another TCP connection is originated as an HTTP / HTTPS client from the Provider Server to the Requestor Server, i.e the physical connection 2. HTTP v1.1 is required [RFC2616].

The physical connection 1 shall carry the service requests from the Requestor Server to the Provider Server and the physical connection 2 the notification requests from the Provider Server to the Requestor Server.

The HTTP / HTTPS transport for SSP requires persistent TCP connection between the servers. HTTP / HTTPS requests and responses are pipelined on the TCP connection. Pipelining allows a HTTP / HTTPS client to make multiple requests without waiting for each response, but the HTTP / HTTPS server must send its responses to those requests in the same order that the requests were received.

The pipelining behavior of the persistent TCP connection may decrease the service provisioning throughput, because one request whose response needs more processing time may block all the other ready responses belonging to later requests. This is the reason why the SSP transaction is separated from the HTTP / HTTPS transaction on the following way shown on the Figure 1.

The SSP transaction request and the reply are delivered only by HTTP / HTTPS POST requests. The SSP request is carried in the HTTP / HTTPS body. The HTTP /HTTPS POST reply is a dummy reply, i.e. the body is empty (status code= OK).

The SSP transaction request initiated by the Requestor Server is transmitted on the physical connection 1, and the response of the same SSP transaction is delivered on the physical connection 2. The transaction identifier associates the two transaction halves.

Similarly the SSP notification transaction request initiated by the Provider Server is transmitted on the physical connection 2, and the response of the same SSP transaction is delivered on the physical connection 1.

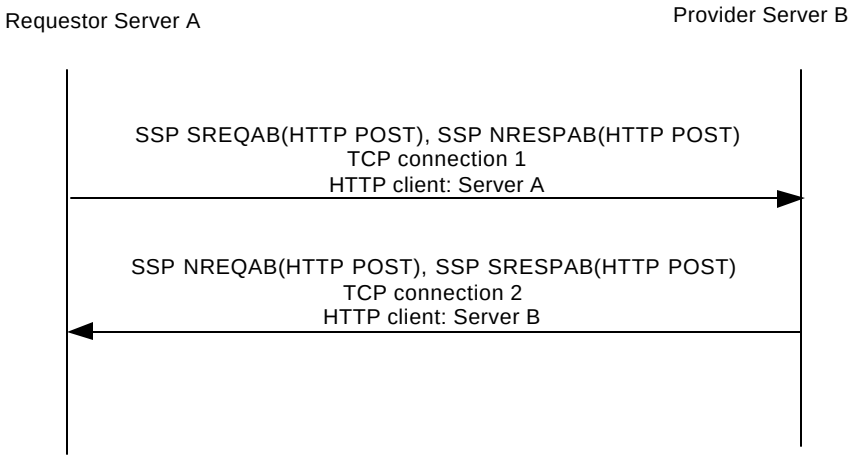

Figure 1. HTTP / HTTPS Binding for One Session Provisioned by Server B

In this example server A is the Requestor Server and server B is the Provider Server.

SREQAB: service request from A to service provider B
NRESPAB: notification response from server A to service provider B

NREQAB: notification request from B to service requester A
SRESPAB: service response from B to service requester A

In this example server A is the Provider Server and server B is the Requestor Server as shown on Figure 2.

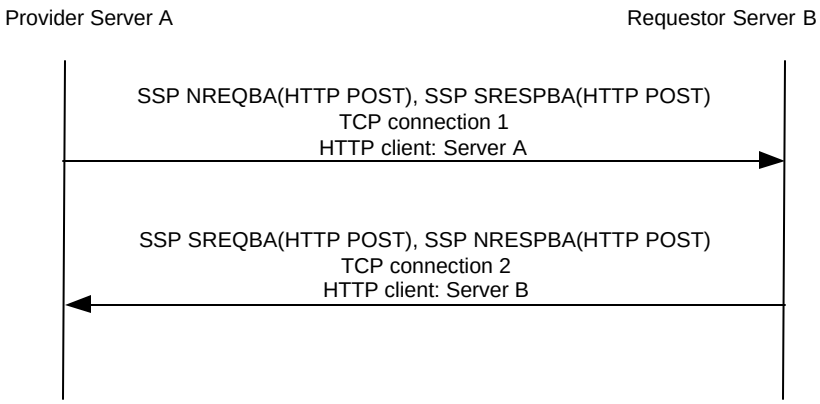

Figure 2. HTTP / HTTPS Binding for the Other Session Provisioned by Server A

where:

SREQBA: service request from B to service provider A
NRESPBA: notification response from server B to service provider A

NREQBA: notification request from A to service requester B
SRESPBA: service response from A to service requester B

3.2. Connection Pair Reuse

If the connection pair is (re)used by the two sessions, the physical connection 1 carries:

- for session 1
  - the SSP service transaction requests from Requestor Server A to Provider Server B
  - the SSP notification responses from Requestor Server A to Provider Server B
- for session 2
  - the SSP service transaction response from Provider Server A to Requestor Server B
  - the SSP notification request from Provider Server A to Requestor Server B

and similarly the physical connection 2 carries:

- for session 1

  - the SSP service transaction response from Provider Server B to Requestor Server A
  - the SSP notification request from Provider Server B to Requestor Server A

- for session 2

  - the SSP service transaction request from Requestor Server B to Provider Server A
  - the SSP notification responses from Requestor Server B to Provider Server A

3.3. Multiple Connection Pairs

Servers may open additional connection pairs belonging to the same session pair if the SSP redirection is allowed.

3.4. SSP Message Content Type

The content type of the SSP message is:

Content-Type: application/vnd.wv.ssp.xml

3.5. HTTP / HTTPS Redirection

The WV domain must understand standard HTTP / HTTPS redirection codes [RFC2616] and associated information headers. HTTP / HTTPS redirection allows WV server to redirect to other servers based on existing load balancer.

HTTP / HTTPS redirection is only allowed in Step 1 and / or Step 3 of the connection establishment, i.e. the first SendSecretToken primitive after the TCP connection is set up.

3.6. Header Extensions for HTTP / HTTPS Binding

The following two headers are extensions for faster dispatching of the SSP messages to spare the XML document parsing.

This header extension must be used to carry the transaction identifier in all HTTP / HTTPS POST request:

```
header       =  x-wv-transactionid ":"  header-value  CRLF
header-value  =  1*alphanum
alphanum = alpha | digit | "_"
```

This header extension must be used to carry the session identifier in all HTTP / HTTPS POST request if the session is established:

```
header       =  x-wv-sessionid ":"  header-value  CRLF
header-value  =  1*alphanum
alphanum = alpha | digit | "_"

alpha    = lowalpha | upalpha

lowalpha = "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" |
           "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" |
           "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"

upalpha  = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" |
           "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" |
           "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"

digit    = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
           "8" | "9"
```

The character "*" preceding an element indicates repetition. The full form is "<n>*element" indicating at least <n> occurrences of element; "1*element" requires at least one.

Elements separated by a bar ("|") are alternatives, e.g., "yes | no" will accept yes or no.